Recursive definition of global cellular-automata mappings

Feldberg, Rasmus; Knudsen, Carsten; Rasmussen, Steen

Published in:
Physical Review E. Statistical, Nonlinear, and Soft Matter Physics

Link to article, DOI:
10.1103/PhysRevE.49.1699

Publication date:
1994

Document Version
Publisher's PDF, also known as Version of record

Recursive definition of global cellular-automata mappings

Rasmus Feldberg,[1] Carsten Knudsen,[1] and Steen Rasmussen[2]
[1]*Physics Department and Center for Modeling, Nonlinear Dynamics and Irreversible Thermodynamics, The Technical University of Denmark, DK-2800 Lyngby, Denmark*
[2]*Analysis Division Office, Section for Simulation Applications, Center for Nonlinear Studies, and Theoretical Division, Complex Systems Group, MS-B258, Los Alamos National Laboratory, Los Alamos, New Mexico 87545 and Santa Fe Institute, 1660 Old Pecos Trail, Santa Fe, New Mexico 87501*
(Received 13 April 1993)

A method for a recursive definition of global cellular-automata mappings is presented. The method is based on a graphical representation of global cellular-automata mappings. For a given cellular-automaton rule the recursive algorithm defines the change of the global cellular-automaton mapping as the number of lattice sites is incremented. A proof of lattice size invariance of global cellular-automata mappings is derived from an approximation to the exact recursive definition. The recursive definitions are applied to calculate the fractal dimension of the set of reachable states and of the set of fixed points of cellular automata on an infinite lattice.

PACS number(s): 05.45.+b

I. INTRODUCTION

The use of cellular automata to simulate the dynamics of physical systems has increased tremendously over the past few years. Although the idea of using discrete methods for modeling partial differential equations is old, the actual proof that cellular-automata techniques can approximate the solutions of hydrodynamic partial differential equations was first given in 1986 [1]. This was shown assuming microscopic conservation laws for energy and momentum of the interaction of virtual particles on cellular-automata lattices. Since then the activity in this particular area of cellular-automata research, which is usually referred to as lattice gases, has been rapidly expanding [2–6].

Cellular automata and related formalisms are also being extensively exploited in another new area of nonlinear science, artificial life [7,8], in an attempt to formulate simple algorithms capable of exhibiting the complex dynamics found in natural systems. This is being done much in the spirit of von Neumann and Ulam's original idea behind the creation of the cellular-automata formalism [9].

Despite the increasing use of cellular-automata simulation techniques and their growing potential for applications, a parallel development of analytical tools has not followed. Only few mathematical results exist in this area; see, for instance, [10–16].

The mathematical results to be presented in this paper are based on a technique derived from lattice size invariance of a particular definition of global cellular-automata mappings. The global mapping is the function relating any state of a cellular automaton to its state at the subsequent time step. Using a recursive technique based on these mappings some results relating the cellular-automata rules to the state space structure are derived. We consider only one-dimensional cellular automata with two possible states of each lattice site and periodic boundary conditions. Each site evolves deterministically according to rules involving the values of the nearest neighbors and the site itself. This class of cellular automata is normally referred to as elementary cellular automata [17]. It should be noted, however, that the methods to be presented are applicable to more complicated cellular automata as well.

The evolution rule of an elementary cellular automaton is normally expressed through a function mapping any possible neighborhood into a corresponding site state:

$$a_i^{t+1}=\phi(a_{i-1}^t a_i^t a_{i+1}^t)\ , \tag{1}$$

where a_i^t is the state (i.e., 0 or 1) of the ith site at time t. ϕ is referred to as the cellular-automaton rule table. There are 256 different elementary cellular automaton rule tables which are referred to by the rule numbers 0 to 255 as in [17]. Some rules are equivalent, either by reflection or by inversion or by a combination of both [18]. Therefore the complete set of rules can be divided into 88 subsets of equivalent rules. Any 88 rules representing all of these subsets will be referred to as a basic set of cellular automaton rules. We will use the basic subset suggested by Li and Packard [18] to represent all elementary cellular automata.

To represent global mappings of cellular automata graphically, an ordering of the set of all possible states needs to be introduced. The state of a cellular automaton with n lattice sites can be represented by the sequence $a=a_1,\ldots,a_n$, where a_i, $i\in\{1,2,\ldots,n\}$, is either 0 or 1. The ordering of the states is the readily given by reading the corresponding sequences as binary numbers.

Let Φ_n denote the global mapping of a cellular automaton with n lattice sites. Φ_n maps a state of a cellular automaton at time t to the state at time $t+1$, according to the rule table ϕ. In Fig. 1 three different Φ_{12} mappings for elementary cellular automata with the rules 12, 129, and 137 are shown. Note that all of the shown mappings apparently have a self-similar structure. Figure 2 shows the global mappings Φ_{10}, Φ_{11}, and Φ_{12} for the rule-137 cellular automaton. Note that the structure of the global

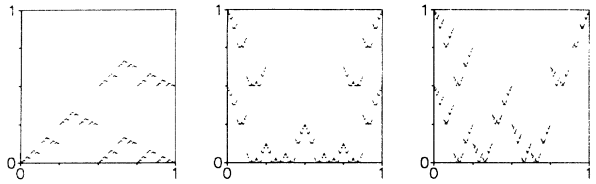


FIG. 1. Global mappings of the 4096 possible states of a lattice of size 12 for cellular-automaton rules 12, 129, and 137 (left to right). The states are mapped onto the unit interval by dividing the corresponding binary numbers by 2^n, where n is the lattice size. Note the apparent self-similar structure of the three mappings.

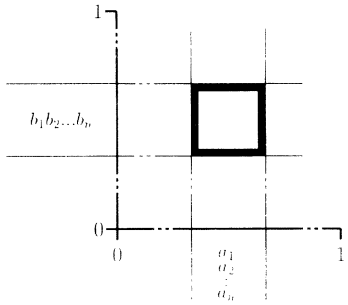


FIG. 3. Transition from one state $a=a_1 \ldots a_n$ of a cellular automaton to another $\Phi_n(a)=b=b_1 \ldots b_n$. The cellular automaton has n lattice sites, hence the side length of the square representing the transition is 2^{-n}.

mapping seems to be independent of the number of sites. Global mapping of cellular automata with different numbers of lattice sites cannot, of course, be identical, since they contain different numbers of points. However, when the number of lattice sites is sufficiently large, the graphical representations of global mappings with varying lattice size all look alike. These properties—the self-similar structure and the lattice size invariance—extend to all elementary cellular automata.

II. RECURSIVE DEFINITION OF GLOBAL MAPPINGS

A. Derivation of recursive box definitions

In the following we investigate the transition from the global mapping of a cellular automaton with n lattice sites to the mapping of one with $n+1$ sites, i.e., we are interested in the mapping **F**: $\Phi_n \mapsto \Phi_{n+1}$.

Consider a cellular automaton with n sites, where $n \geq 2$. In a graphical representation of the mapping the axes, given the length 1, can be divided into 2^n intervals, each representing a state of the cellular automaton. In this way the transition from one state $a=a_1 \ldots a_n$ of the cellular automaton to another $\Phi_n(a)=b=b_1 \ldots b_n$, is represented by a square with side length 2^{-n}, as illustrated in Fig. 3.

As a new site is added to the cellular automaton, the number of possible states is doubled and the side lengths of the squares representing state transitions are halved. In the following the squares with side length 2^{-n} are referred to as *boxes* and the squares with side length $2^{-(n+1)}$ are referred to as *sub-boxes*. Due to the chosen ordering of the states, the interval on the abscissa axis that contained a before the addition of the new site now contains two sequences $a^0=a_1 \ldots a_n 0$ and $a^1=a_1 \ldots a_n 1$. The question is how the addition of the new site affects the *vertical* position of the two sub-boxes, which appear instead of the original box. Denote the sequences, representing the states that a^0 and a^1 maps into, by $\Phi_{n+1}(a^0)=b^0=b_1^0 \ldots b_n^0 b_{n+1}^0$ and $\Phi_{n+1}(a^1)=b^1=b_1^1 \ldots b_n^1 b_{n+1}^1$, respectively. Figure 4 illustrates the transitions from a^0 to b^0 and from a^1 to b^1 after the addition of the new lattice site.

The neighborhoods in a^0 and a^1 leading to $b_2^0 \ldots b_{n-1}^0$ and $b_2^1 \ldots b_{n-1}^1$ are left unchanged by the new lattice site, implying that these subsequences are identical to $b_2 \ldots b_{n-1}$. Only b_1, b_n, and b_{n+1} are affected by the new lattice site. The values of b_1, b_n, and b_{n+1} depend on a_1, a_2, a_{n-1}, a_n, and on the value of the new lattice site a_{n+1}. Given the values of a_1, a_2, a_{n-1}, and a_n for a given box, one can therefore calculate the positions of the two sub-boxes that are created from the original box as the extra lattice site is added to the cellular automaton. Since there are only 16 possible combinations of a_1, a_2, a_{n-1}, and a_n, there are at most 16 different types of behavior of the boxes in an elementary cellular au-

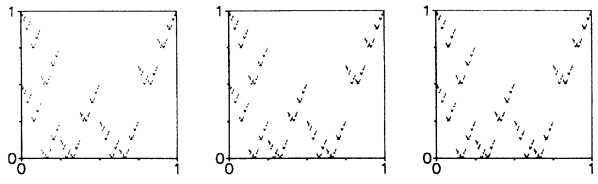


FIG. 2. Global mappings of the cellular automata with rule 137, for lattice sizes 10, 11, and 12 (left to right). The states are mapped onto the unit interval by dividing the corresponding binary numbers by 2^n, where n is the lattice size. Note the apparent size invariance of the structure of the mappings.

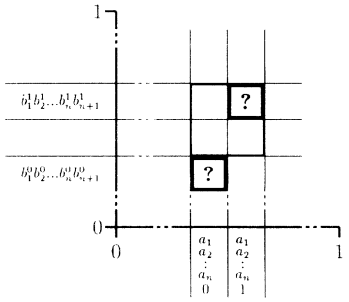


FIG. 4. The transition shown in Fig. 3 after the addition of a new lattice site to the automaton. The abscissa axis interval containing a in Fig. 3 contains the sequences $a^0=a_1 \ldots a_n 0$ and $a^1=a_1 \ldots a_n 1$ after the addition of the new site. The question is how the new site affects the vertical position of the two new sub-boxes with side length $2^{-(n+1)}$ as compared to the position of the original box with side length 2^{-n}.

tomaton of any lattice size. We name these different boxes A through P, so that A corresponds to boxes with $a_1a_2a_{n-1}a_n=0000$, B corresponds to boxes with $a_1a_2a_{n-1}=0001$, C corresponds to boxes with $a_1a_2a_{n-1}a_n=0010$, and so on.

The limited difference between b, b^0, and b^1 implies that the possible vertical positions of sub-boxes, as compared to the original box, can be summarized in the following way.

(i) Changes of b_1 lead to a vertical shift of the length $\frac{1}{2}$, corresponding to 2^{n-1} boxes since b_1 is the most significant bit of b. These shifts will be referred to as *long shifts*. The direction of the shift depends on the change of b_1. If b_1 changes from 0 to 1, the shift is upwards; if it changes from 1 to 0, it is downwards. If b_1 does not change, no long shift takes place.

(ii) Changes of b_n lead to a vertical shift of the length 2^{-n}, corresponding to one box since b_n is the least significant bit of b, before the addition of the new site. These shifts will be referred to as *short shifts*. The direction of the shift depends on the change of b_n in the same way as the shifts resulting from changes of b_1.

(iii) The values of b^0_{n+1} and b^1_{n+1} determine the vertical position of the corresponding sub-box within the box reached after any shifts caused by changes of b_1 or b_n. Since the side length of a sub-box is half the side length of an original box, a sub-box can be positioned either in the upper or in the lower half of a box, depending on b^0_{n+1} or b^1_{n+1}.

To illustrate the above algorithm we compute the positions and the types of sub-boxes appearing from a type D box in a cellular automaton under the action of rule 137. The rule table ϕ corresponding to this rule is shown in Table I.

A type D box has $a_1a_2a_{n-1}a_n=0011$. Before the addition of a new lattice site to the cellular automaton $b_1=\phi(a_na_1a_2)=0$ and $b_n=\phi(a_{n-1}a_na_1)=0$. After the addition of the new site $b^0_1=\phi(0a_1a_2)=1$, $b^0_n=\phi(a_{n-1}a_n0)=0$, $b^0_{n+1}=\phi(a_n0a_1)=0$, $b^2_1=\phi(1a_1a_2)=0$, $b^1_n=\phi(a_{n-1}a_n1)=1$, and $b^1_{n+1}=\phi(a_n1a_1)=0$.

Considering the left sub-box corresponding to $b^0=\Phi_{n+1}(a^0)$ first, it is seen that this sub-box is shifted 2^{n-1} boxes upwards in a long shift, because b_1 changes from 0 to 1 and b_n is unchanged. Since $b^0_{n+1}=0$, the sub-box is placed in the bottom of the box reached after the vertical shift. The two last bits in the sequence appearing when a 0 is added after a_n are 10. The first two bits in the sequence are unchanged as the new site is added, since it is assumed that $n\geq 2$. Thus the left sub-box becomes $a_1a_2a_na_{n+1}=0010$, corresponding to a sub-box of type C.

The right sub-box is shifted one box upwards in a short shift, since b_1 is unchanged and b_n changes from 0 to $b^1_n=1$. The new site b^1_{n+1} has the value 0, so the sub-box is placed in the bottom of the box reached after the vertical shift. It is easily seen that the right sub-box becomes a box of type D.

In order to describe the behavior of a particular type of box in a simple and clear manner, we introduce the box definition shown in Fig. 5. This definition consists of a square divided into four subsquares. Two of the subsquares contain information about a sub-box, one in the left-hand side of the square and one in the right-hand side. The vertical position of a nonempty subsquare indicates the final position of the sub-box inside the box reached after the vertical shifts. The information inside the two nonempty subsquares consists of up to three symbols. The symbol in the middle indicates the type of the corresponding sub-box and is one of the letters A through P. The symbol, if any, to the left of the type indicator indicates long shifts of the corresponding sub-box. ⇑ indicates long shifts upwards and ⇓ indicates long shifts downwards. The symbol, if any, to the right of the type indicator indicates short shifts of the corresponding sub-box. ↑ indicates short shifts upwards and ↓ indicates short shifts downwards.

Figure 6 defines the mapping $\mathbf{F}$: $\Phi_n \mapsto \Phi_{n+1}$ for rule 137 through the 16 box definitions. The derivation of the box definitions can be done systematically and a catalogue of the box definition of 88 basic cellular automata rules can be found in [19].

We have now shown how to derive the set of box definitions characterizing a given cellular-automaton rule. The set of box definitions constitutes a lattice-size-independent iteration algorithm, by which it is possible to construct the global mapping of any size cellular automaton, by incrementing the number of lattice sites the ap-

TABLE I. The rule table ϕ of rule 137.

$a_{i-1}a_ia_{i+1}$	$\phi(a_{i-1}a_ia_{i+1})$
000	1
001	0
010	0
011	1
100	0
101	0
110	0
111	1

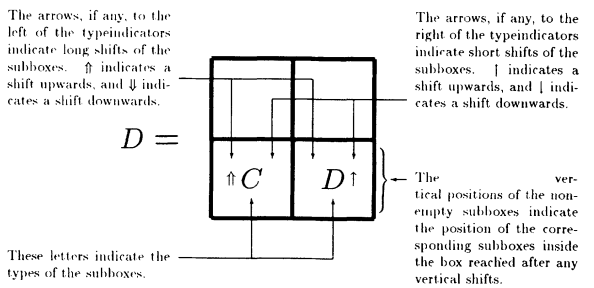


FIG. 5. Definition of a box for a recursive definition of the global mapping of a cellular automaton. The figure illustrates the behavior of a type-D box at the addition of a new site to the cellular automaton. The box is split into two sub-boxes, one of which is shifted 2^{n-1} boxes upwards and placed in the bottom of the box reached after the shift. The type of this sub-box is C. The other sub-box is shifted one box upwards and placed in the bottom of the box reached after the shift. This sub-box has the type D as has the original box.

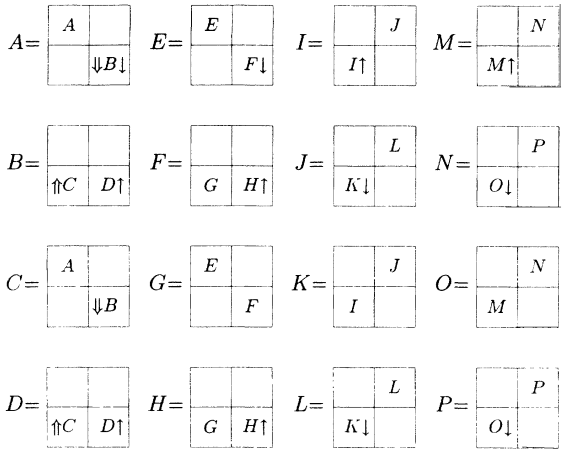


FIG. 6. Complete set of box definitions for the cellular automaton with rule 137. The notation demonstrated in Fig. 5 is used. Note that some of the box definitions are identical, for instance, *B* and *D*.

propriate number of times.

As an illustration of the method the recursive construction of the global mapping for a cellular automaton under the action of rule 137 follows. One starts with a lattice of size two. From the rule table it is easily seen that $\Phi_2(00)=11$, $\Phi_2(01)=00$, $\Phi_2(10)=00$, and $\Phi_2(11)=11$. To determine the types of the four boxes in the global map for $n=2$ the sequences at the ordinate axis are inspected. For this particular size of the cellular automaton the two first bits of a sequence are identical to the two last bits. Therefore, the types of the boxes for the global mapping are, from left to right, *A*, *F*, *K*, and *P*. This result is true not only for rule 137, but for all elementary cellular automata, due to the ordering of the possible states of the automaton. Figure 7(a) shows the global mapping of the cellular automaton with two sites and with the box types indicated. Using the complete set of box definitions from Fig. 6 it is easy to iterate the mapping and thereby obtain the global mapping for the cellular automaton of any lattice size. The mappings for three and four site cellular automata are shown in Figs. 7(b) and 7(c).

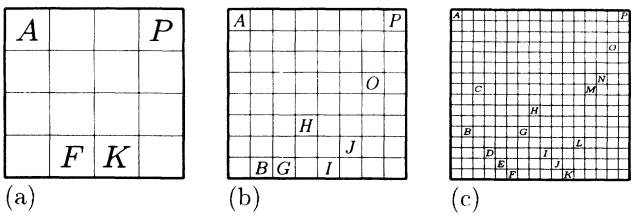


FIG. 7. The global map of the cellular automaton with rule 137 on (a) two lattice sites, (b) three lattice sites, and (c) four lattice sites. The types of the boxes in the mapping are indicated within each box. Note that all 16 box types can at earliest be found in the automaton with four lattice sites.

B. Computational aspects of the recursive mapping definition

It should be noted that the computational resources involved in the updating of a cellular automaton on a lattice of size n using this recursive construction scheme is of the same order of magnitude as the work required to propagate the local rule on the lattice of size n in the conventional way.

By the conventional method, n cells need to be updated, which requires $O(n)$ operations, i.e., the work grows linearly with n.

To update the state $a=a_1a_2\ldots a_n$ for a cellular automaton into the corresponding state $b=\Phi_n(a)$ using the recursive definition of Φ_n one starts out with $b_1b_2=\Phi_2(a_1a_2)$, which hardly represents any computational obstacle. From that point on a stepwise adding of the remaining bits to a is done, using the box definitions to keep track of the bits, i.e., the vertical position, of b. This also requires of the order of $O(n)$ operations.

The main advantage of the recursive definition of global cellular-automata mappings, however, does not lie in its updating scheme but in the information it offers about the global state space structure. In Secs. II C, III A, V A, and V B we shall take advantage of this.

C. Self-similarity of the mappings

The apparent self-similarity of Figs. 1 and 2 can be explained from the recursive definition of the global mapping. We have already shown that a given global mapping can be constructed from no more than 16 different types of boxes. The graphical representation of the mapping can be divided into four equally large areas or columns, in the following referred to as the four *main columns*. Due to the chosen ordering of states, the sequences in the intervals of the abscissa axis corresponding to the four main columns starts with $00\ldots,01\ldots,10\ldots$, and $11\ldots$, respectively. Considering a given main column, the number of different types of boxes that can appear inside the column is four, since the first two bits in all sequences inside a main column are fixed. For instance, the box types that can occur in the first main column are *A*, *B*, *C*, and *D*. Furthermore, a given box in a main column will have any of the four box types appearing in the main column as a sub-box after two additions of new sites. This is due to the fact that by the addition of new sites, i.e., new bits in the sequences representing the states of a cellular automaton, all combinations of the two last bits in the sequences representing subboxes of a box will appear. For instance, according to the definition in Fig. 6, a type-*A* box will turn into a type-*A* box and a type-*B* box, once it is iterated. In the next iteration of the type-*B* box, this box will turn into a type-*C* box and a type-*D* box.

An arbitrary box in one of the main columns of a global mapping of a cellular automaton with n sites will generate a certain structure of the global mapping as n is repeatedly incremented. Since a box will have any of the box types appearing in the main column, including itself, as a sub-box after one or two additions of new sites, the structure generated by the box will be generated repeat-

edly in smaller and smaller versions. This demonstrates that a structural self-similarity is present in the iterative process involving the different types of boxes corresponding to a particular cellular-automata rule. However, the structural self-similarity does not necessarily imply geometric self-similarity of the global mappings, as indicated by the figures. With the introduction of local box definitions in Sec. III, it will become clear that the self-similarity extends to the geometric representation of the global mappings.

III. SIMPLIFIED RECURSIVE DEFINITION OF THE MAPPINGS

The recursive box definitions presented in Sec. II have the property that boxes are allowed to place sub-boxes outside themselves. This is needed for the boxes to generate an exact representation of the global mapping of any cellular automaton with any number of sites. However, the possibility of these shifts of sub-boxes makes the box definitions rather complicated as compared to the new type of box definitions, which will be introduced in the following. These new types of box definitions are referred to as *local box definitions,* since no long or short shifts will be allowed.

The local box definitions have the advantage that they can be generated by hand, simply by inspecting a global mapping represented graphically with a reasonably good resolution. This graphical method, which will be demonstrated in Sec. IV, shows how accurately the local box definitions reflect the structure of the global mappings of cellular automata. However, the local box definitions can also be derived systematically from the nonlocal box definitions. It is also possible to generate an exact global mapping from the local box definitions by using certain complementary selection rules to be presented in Sec. IV A. However, the approximations of the global mappings that can be generated from the local box definitions have certain properties in common with the real global mappings. Therefore we can take advantage of the simplicity of the local box definitions to extract information about cellular automata that would be hard to obtain using other techniques.

Let us start by establishing a formal basis for the local box definitions. Recall that Φ_n treated as a set consists of 2^n boxes placed in a $2^n \times 2^n$ matrix and that Φ_n can be generated using the non-local box definitions (Sec. II). Now define another set of boxes Ψ_n, in the $2^n \times 2^n$ matrix. We require from Ψ_n that (i) it can be constructed from a set of local box definitions, (ii) it contains Φ_n as a subset for all n, and (iii) it is a minimal construction in the sense that it does not contain any boxes not needed in order to comply with (i) and (ii).

Consider a column in the $2^n \times 2^n$ matrix of Φ_n. Such a column contains exactly one box, thus the same column in Ψ_n must contain a box at the same position. Suppose that the box in Φ_n makes a shift, long or short, of one of its sub-boxes. In that case, Φ_{n+1} will have a sub-box outside the original box, hence Ψ_{n+1} must also have a sub-box outside the original box. Since Ψ_{n+1} is to be generated from Ψ_n by the use of local box definitions, Ψ_n also has to include a box containing the sub-box outside the original box of Φ_n. It readily follows that Ψ_n has to contain all sub-boxes that ever appear as Φ_n is iterated repeatedly by F: $\Phi_n \mapsto \Phi_{n+1}$.

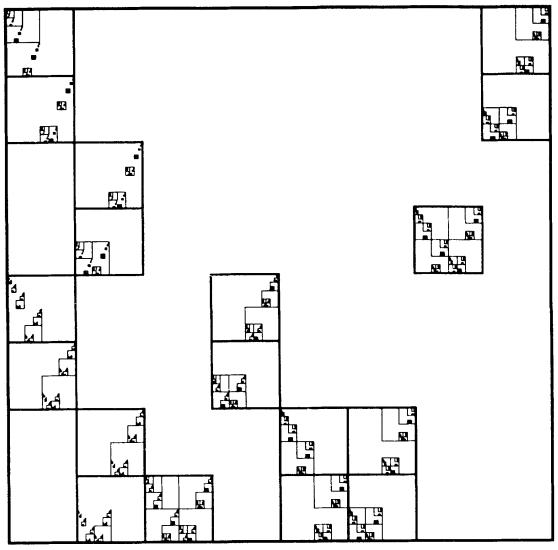


FIG. 8. Ψ_3 (thick lines) and $\bigcup_{i=3}^{\infty}\Phi_i$ (thin lines) for rule 137. Note that the boxes of Ψ_3 with the side length $\frac{1}{8}$ exactly covers $\bigcup_{i=3}^{\infty}\Phi_i$, i.e., there is no box from Ψ_3 that does not contain a box from $\bigcup_{i=3}^{\infty}\Phi_i$ and all parts of $\bigcup_{i=3}^{\infty}\Phi_i$ are covered by Ψ_3.

This leads to the following definition of Ψ_n.

Definition. Ψ_n is the set of boxes with side length 2^{-n} arranged inside a $2^n \times 2^n$ matrix that *exactly covers* $\bigcup_{i=n}^{\infty}\Phi_i$; exactly covers meaning that there is no box from Ψ_n that does not contain one or more boxes from $\bigcup_{i=n}^{\infty}\Phi_i$ and that all boxes from $\bigcup_{i=n}^{\infty}\Phi_i$ are covered by Ψ_n.

In Fig. 8, $\bigcup_{i=3}^{\infty}\Phi_i$ (thin lines) is shown together with Ψ_3 (thick lines) in order to illustrate the definition of Ψ_n for rule 137.

A. Lattice size invariance of the mappings

Consider a column c—not necessarily a main column—of width 2^{-n} in Ψ_n and Φ_n. Φ_n has exactly one box inside c. If the box is ever to place sub-boxes outside itself in *short shifts,* this has to take place in the first addition of a new site to the cellular automaton, since this is the only time the neighborhood leading to b_n is changed. Short shifts may occur at subsequent site additions, but by then the increased resolution of the mapping will prevent the sub-boxes from getting outside the box they landed in by the first site addition.

If there are any *long shifts* in any of the box types of the main column in which c is placed (or is identical to), a long shift will take place already at the first addition of a new site, and thus at all subsequent additions. This is due to the fact that both of the possible neighborhoods leading to b_1, i.e., $(0a_1a_2)$ and $(1a_1a_2)$, occur as the first new site is added. If long shifts are present either $b_1^0 = \phi(0a_1a_2)$ or $b_1^1 = \phi(1a_1a_2)$ has to differ from the

original most significant bit $b_1=\phi(a_n a_1 a_2)$. It follows readily that at least one sub-box of any box does *not* make a long shift. A similar result applies to short shifts.

The possible scenarios by which a column of Ψ_n can be populated with boxes in order to exactly cover $\bigcup_{i=n}^{\infty}\Phi_n$ can now be outlined.

(i) In the case that there are no long shifts and no short shifts take place in the first addition of a new site, there will never appear sub-boxes outside the original box. Thus Ψ_n contains one box in c at the same position as the box in Φ_n.

(ii) In the case that there are no long shifts and a short shift takes place in the first addition of a new site, this will leave one sub-box inside the original box and one sub-box in the box right below or right above the original box. In all subsequent site additions, the sub-boxes will remain inside these two boxes of Φ_n. Thus Ψ_n contains two boxes in c positioned atop of each other with one of them at the same position as the box in Φ_n.

(iii) In the case that there are long shifts in the main column and no short shift takes place in the first site addition, a result corresponding to the short shift case (ii) applies, except that the two boxes in Ψ_n are placed 2^{n-1} boxes from each other corresponding to half the length of the ordinate axis.

(iv) In the case that a column has long shifts as well as a short shift in the first site addition, two different scenarios are possible. Either one sub-box makes a short shift while the other makes a long shift, or one sub-box makes both a long and a short shift (in which case the two shifts are superimposed) while the other sub-box remains inside the original box in the first site addition. In any case the long shifts resulting from the subsequent addition of a new site implies that the c has to contain four boxes in Ψ_n. An interesting implication of cases (iii) and (iv) is that if a main column has long shifts, the part of Ψ_n that is inside the main column will consist of two identical images shifted 2^{n-1} boxes vertically, corresponding to half the length of the ordinate axis. In other words, the upper and the lower half of Ψ_n will be identical inside the main column; see, for instance, Fig. 1.

All four scenarios are sketched in Fig. 9. The above discussion of scenarios (i)–(iv) reveals that the maximum number of boxes in any column of Ψ_n is four, independent of n and the cellular-automaton rule number. In other words, the number of boxes in Ψ_n is less than or equal to four times the number of boxes in Φ_n. Obviously Ψ_n contains at least as many boxes as Φ_n. Thus the fractal dimension d_c (i.e., the capacity dimension [20]) of Ψ_n, as n tends to infinity, equals that of Φ_n, namely 1.

The discussion also reveals that in a column where scenarios (i)–(iii) take place, Φ_{n+1} will have boxes inside all boxes of Ψ_n. In a column where scenario (iv) takes place Φ_{n+1} will not have boxes inside all boxes of Ψ_n, but Φ_{n+2} will. It is easily seen that if Φ_{n+i}, where $i\geq 1$, has boxes inside all boxes of Ψ_n, all Φ_{n+j}, where $j\in[i;\infty[$ will have boxes inside Ψ_n. In other words, for all $n\geq 2$, all of $\Phi_{n+2},\Phi_{n+3},\ldots,\Phi_\infty$ has boxes inside all boxes of Ψ_n.

This accounts for the apparent lattice size invariance observed in Fig. 1. Since any graphical representation of a global mapping has a finite resolution (not to mention the resolution of the human eye) corresponding to a box with side length $2^{-\text{res}}$, any graphical representation of Φ_n, where $n\geq \text{res}+2$ is identical to Ψ_{res}. In the following section we will take advantage of this fact in the construction of local box definitions directly from graphical representations of global cellular-automata mappings.

IV. DERIVATION OF LOCAL BOX DEFINITIONS FROM GRAPHICAL REPRESENTATIONS OF THE MAPPINGS

The simplified recursive definition of the global mapping that we have just described formally in Sec. III can be derived in at least two ways. First, it can be derived from the nonlocal box definitions [19]; the algorithm for the derivation can be found in the Appendix. Second, it can be derived in a more direct and quite elegant way from graphical representations of global mappings. This will be demonstrated in the following.

Contemplate one of the graphical representations of global mappings shown in Fig. 2. The formation of a set

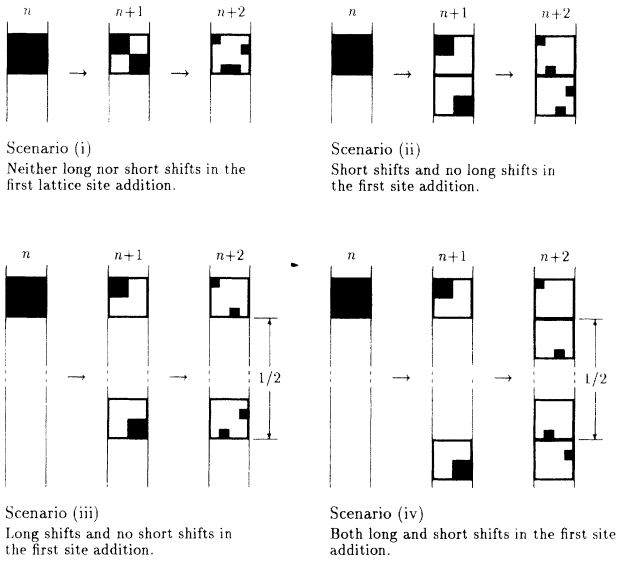


FIG. 9. Different scenarios for the distribution of sub-boxes in a column of Φ_n and the effect of this on Ψ_n. In each scenario the black squares represents Φ_n, Φ_{n+1}, and Φ_{n+2}. The frames indicates the positions of the boxes of Ψ_n. These boxes are supposed to cover the boxes of $\bigcup_{i=n}^{\infty}\Phi_i$ and they are added in each scenario when they are needed in order to cover boxes of Φ_i not already covered. In scenario (i) a single box of Ψ_n is able to cover both Φ_n, Φ_{n+1}, and Φ_{n+2}; due to the rules governing the nonlocal box definitions it therefore also covers $\bigcup_{i=n}^{\infty}\Phi_i$. In scenarios (ii) and (iii) an additional box of Ψ_n is necessary in order to cover $\Phi_n,\Phi_{n+1},\Phi_{n+2}$, and thus $\bigcup_{i=n}^{\infty}\Phi_i$. In scenario (iv) the first site addition results in two sub-boxes located $2^{n-1}+1$ boxes from each other (i.e., a long and a short shift superimposed), which can be covered by two boxes of Ψ_n. Due to long shifts in the subsequent site addition yet another two boxes of Ψ_n are necessary to cover Φ_n,Φ_{n+1}, and Φ_{n+2}. The scenarios in the figure corresponds to the box types *G*, *F*, *C*, and *A*, respectively, from Fig. 6.

of local box definitions capable of generating an image identical to the graphical representation is now presented. The first box to be defined is the box corresponding to the entire image. The box, which is given the name **A**, is divided into four sub-boxes, as shown in Fig. 10(a). There are four nonempty sub-boxes inside **A**. This is in contrast to nonlocal box definitions in which only two sub-boxes are defined in each box, one in each side of the box. Since the picture now being generated will contain up to four boxes per column, there is nothing "illegal" about two sub-boxes on top of each other, and hence four sub-boxes inside one box. The sub-boxes of **A** are named **B** through **E** in a clockwise manner, so that **B** is in the upper left corner of **A**, as shown in Fig. 10(a). Since no shifts are allowed in the definition of a local sub-box, only the new box type has to be specified.

To define **B** the upper left sub-box of **A** is enlarged and divided into sub-boxes as shown in Fig. 10(b). In this case the two sub-boxes on the right-hand side are both empty. By comparing the upper left sub-box of **B** with the upper left sub-box of **A** in Fig. 10(a), it is noted that these two sub-boxes are identical. The same goes for the lower left sub-box of **B** and the upper right sub-box of **A**. Since the two sub-boxes in **A** are already defined as **B** and **C**, respectively, the same names for the sub-boxes of **B** are used. Hence the definition of **B** is that the upper left sub-box is a type-**B** box and the lower left subbox is a type-**C** box, while the other two sub-boxes are empty.

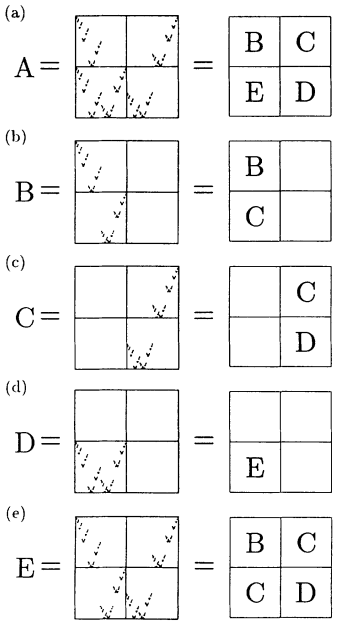


FIG. 10. Derivation of local box definitions of a rule-137 cellular automaton. To derive a set of local box definitions of a global cellular-automaton mapping, the entire image of the mapping is considered to be a local box definition. That is, a name is assigned to it and it is divided into four sub-boxes, each of which is given a name as well. If the image of a sub-box is recognized as an already defined box or sub-box, the name of that box is given to it, otherwise a new name is assigned to the subbox. This procedure continues with each of the sub-boxes, until all boxes, sub-boxes, and *sub*-sub-boxes are defined. This will happen within a limited number of steps due to the self-similarity of the global mapping.

In the enlargement of **C** shown in Fig. 10(c) it is easily seen that **C** contains two empty sub-boxes on the left-hand side. The sub-boxes in the right-hand side are recognized as **C** and **D** by inspecting Fig. 10(a). The definitions of **D** and **E** are found in a similar manner, and they are shown in Figs. 10(d) and 10(e).

In general, the derivation of local box definitions from a graphical representation of the global mapping is carried out by repeatedly dividing the undefined sub-boxes into new sub-boxes until all boxes are recognized as previously defined boxes. By systematic use of the algorithm in the appendix, one can show that this always takes place within a limited number of box definitions, no more than seven.

Note that none of the definitions of sub-boxes of **A** required definitions of new box types. This need not always to be the case. However, for all elementary cellular-automata rules, all box types needed to define Ψ_n will be encountered within the first three levels of boxes, corresponding to $n \leq 3$.

Note also that the main box **A** does not reappear as a sub-box as new sites are added to the cellular automaton. It is a transient box type. In some cases (for instance, rules 72, 104, and 132) also the immediate sub-boxes of the main box are transient box types that only appear once as the number of sites is repeatedly increased. This apparent disagreement with the nonlocal box definitions, in which all box types reappear infinitely many times as new sites are added, is explained by the fact that the main box and its immediate sub-boxes correspond to 2^0 and 2^1 possible states of the cellular automata, whereas the nonlocal box definitions are only valid for 2^2 and more possible states.

One may ask if it is always straightforward to decide if two sub-boxes are identical or not. For instance, would it be possible for a difference between two sub-box images to be of a magnitude hardly visible? This would not be possible, since differences between images are the result of differences between sets of nonlocal box definitions. Thus any difference has to be of a magnitude corresponding to 2^{-3} of the side length of an image, i.e., hardly possible to overlook. This is due to the fact that all nonlocal box types appearing in a main column appear after at most two site additions, and thus are evaluated in the third site addition.

The entire set of local box definitions of rule 137 consists of five boxes. This number may vary from rule to rule. For instance, rules 0 and 204 (the zero-fixed-point rule and the identity rule, respectively) has only one box in their definitions, while rules such as 22 and 90 has seven different boxes.

In a sense the number of local box definitions needed to construct Ψ_n of a cellular automaton reflects the complexity of Ψ_n and thus the global mapping [21,22]. However, a classification of cellular-automata rules according to the number of local box definitions needed to produce Ψ_n does not resemble Li and Packard's classification [19,18]. This indicates that the number of local boxes needed to construct the global mapping only poorly

reflects the typical steady-state dynamics, which is the basis of Li and Packard's classification and vice versa.

Once given the local box definitions corresponding to a rule, the construction of Ψ_n is straightforward. Start out with the main box **A**, which corresponds to Ψ_0 and evaluate it according to its definition by substituting it with its four sub-boxes. This process is referred to as an iteration of **A**. This should not be confused with an iteration of a cellular automaton. An iteration of a local box definition corresponds to the addition of a site to a cellular automaton. The first iteration of **A** results in Ψ_1, and by repeated iteration of the sub-boxes herein one obtains Ψ_2 and so on. The process is shown in Fig. 11 for Ψ_0 through Ψ_4 for rule 137.

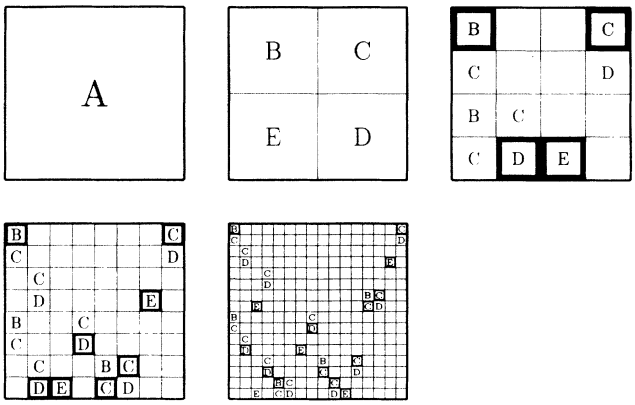


FIG. 11. The first iterations of the set of local box definitions of a rule-137 cellular automaton. Top row: Ψ_0 through Ψ_2 (left to right). Bottom row: Ψ_3 and Ψ_4. The positions of the boxes of Φ_2, Φ_3, and Φ_4 are indicated in Ψ_2, Ψ_3, and Ψ_4, respectively, with bold frames. Note that the number of boxes in each column of Ψ_n, $n \in \{0, \ldots, 4\}$ is 1, 2, or 4.

A. Derivation of the mapping from the approximation

Having derived the local box definitions directly from the graphical representation of the global mappings as described above, the approximate global mapping Ψ_n can be constructed using the local box definitions. It is possible to extract the *exact* global mapping Φ_n of a cellular automaton from Ψ_n using some rules describing how to select the box representing Φ_n in a given column of Ψ_n. To do this one has to consult the nonlocal box definitions, and thus some of the captivating simplicity of the local box definitions is lost. It should therefore be emphasized that the potential of local box definitions lies mainly in utilizing their simplicity to calculate properties of Ψ_n that are shared with Φ_n (Sec. V A) and not in using them to generate the exact global mapping. However, the selection rules can be useful when calculating properties of Φ_n by adjusting results obtained for Ψ_n (Sec. V B).

Consider a column c in Ψ_n. In case of only one box in c, this box is naturally identical to the box in Φ_n. In the case that c contains two or four boxes, the two first and the two last bits of the sequence a, representing c on the abscissa axis, have to be inspected. Remember, that only 16 different combinations of these bits characterizing c exist, and thus at most 16 different selection rules is needed to choose the correct boxes in all of Ψ_n. By investigating how the box in Φ_n distributes sub-boxes into c, it is possible to decide the position of the box to select among the boxes of Ψ_n. As an example consider a type-A box of the rule-137 cellular automaton (see Fig. 6). This box shifts its right sub-box downwards on both a long and a short shift in the first site addition. Due to the long shifts taking place at the subsequent site additions it is easily seen that the original type-A box ends up as the top box of the four nonempty boxes in Ψ_n. Thus the selection rule of any column, which has the characteristic bits cor-

TABLE II. Selection rules for a rule-137 cellular automaton. In the case of rule 137 it is always the top or bottom box of Ψ_n that represents Φ_n when there are four boxes in a column of Ψ_n. This is not the case for all rules.

Characteristic bits	Corresponding type of box	Selection rule	Boxes in column of Ψ_n
0000	*A*	Top box	4
0001	*B*	bottom box	4
0010	*C*	top box	2
0011	*D*	bottom box	4
0100	*E*	top box	2
0101	*F*	bottom box	2
0110	*G*		1
0111	*H*	bottom box	2
1000	*I*	bottom box	2
1001	*J*	top box	2
1010	*K*		1
1011	*L*	top box	2
1100	*M*	bottom box	2
1101	*N*	top box	2
1110	*O*		1
1111	*P*	top box	2

responding to a type-A box in a rule-137 cellular automaton, is that the top box in a column of Ψ_n represents the box in Φ_n.

The entire set of selection rules of a rule-137 cellular automaton is shown in Table II together with the number of boxes in the columns of Ψ_n. In the case of rule 137 it is always the top or the bottom box that represents Φ_n when four boxes are present in a column of Ψ_n. However, this need not always be the case.

V. PROPERTIES OF THE MAPPINGS IN THE CONTINUUM LIMIT

A. Dimension of the set of reachable states

In Secs. II–IV it was demonstrated how to derive recursive box definitions for a given cellular automaton and it was shown how to calculate the global mapping and approximations of the global mapping of a cellular automaton from these box definitions. In Secs. V A and V B this way of defining global mappings and their approximations is used to extract information about the global state space structure of cellular automata.

A characteristic quantity of cellular automata is the number of reachable states in comparison to the total number of states. The set of reachable states can in a graphical representation of a global mapping be identified as the projection of Φ_n, $P(\Phi_n)$, onto the ordinate axis. The dimension d_c of this projection can be calculated using the expression $d_c=\lim_{n\to\infty}\log[N(n)]/\log[1/\epsilon(n)]$, where $N(n)$ is the number of boxes of size $\epsilon(n)$ needed to cover $P(\Phi_n)$. Let $N(n)$ be the number of nonempty rows in Φ_n and let $\epsilon(n)=2^{-n}$. Due the nonlocality of the box definitions of Φ_n it is rather cumbersome to find an expression for $N(n)$. Since it is shown (Sec. III A) that $\Psi_{n-2}\supset\Phi_n$ and that Ψ_{n-2} does not contain any boxes not covering one or more boxes of Φ_n, it readily follows that $P(\Phi_n)$ can be covered by $M(n-2)$ boxes of the size 2^{n-2}, where $M(n-2)$ is the number of nonempty rows of Ψ_{n-2}. Thus $d_c(P(\Psi_\infty))$ equals $d_c(P(\Phi_\infty))$.

Now consider the nonlinear rule 129, which produces chaotic dynamics [18]. The local box definitions of rule 129 are shown in Fig. 12, and the global mapping can be seen in Fig. 1. Obviously Ψ_0 has only one nonempty row (and column) containing the initial **A** box. As **A** is iterated it generates two rows containing **BC** and **ED**, respectively, in the projection. One more iteration yields the rows **BC**, **FG**, **BC**, and **FDEG** in Ψ_2. Yet another iteration results in the rows **BC**, **FG**, **ED**, **BC**, **FG**, **CB**, and **DEGFDE** in Ψ_3. To obtain the projection of Ψ_n on the ordinate axis, the order and the number of the boxes in a row does not matter. The rows of Ψ_3 can therefore be rewritten into $3\times$**BC**, $2\times$**FG**, **DE**, and **DEFG**, where the multiplication sign is used to indicate the number of appearances of a given row. Note that no new types of rows are introduced as Ψ_2 is iterated, and thus will not appear in subsequent iterations.

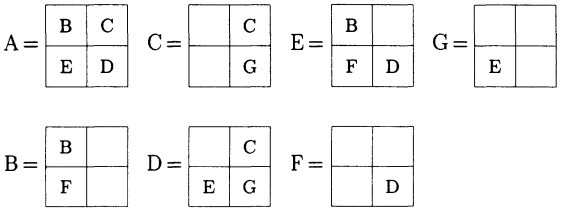


FIG. 12. Complete set of local box definitions for the rule-129 cellular automaton.

By inspecting the local box definitions in Fig. 12, the following rules can now be established for what rows a given row leads to as it is iterated: **BC**→**BC**,**FG**; **DE**→**BC**,**DEFG**; **FG**→**DE**; and **DEFG**→**BC**,**DEFG**. The number of rows as Ψ_n is iterated can now be expressed in matrix form as

$$\begin{bmatrix}\mathbf{BC}_{n+1}\\ \mathbf{DE}_{n+1}\\ \mathbf{FG}_{n+1}\\ \mathbf{DEFG}_{n+1}\end{bmatrix}=\begin{bmatrix}1&1&0&1\\0&0&1&0\\1&0&0&0\\0&1&0&1\end{bmatrix}\begin{bmatrix}\mathbf{BC}_n\\ \mathbf{DE}_n\\ \mathbf{FG}_n\\ \mathbf{DEFG}_n\end{bmatrix}, \tag{2}$$

where $\mathbf{BC}_n$ is the number of rows containing **BC** in Ψ_n, etc.

The total number of nonempty rows in Ψ_n tends to $k\lambda^n$ as n tends to infinity, where k is a positive constant and λ is the largest eigenvalue of the matrix in (2). Since the characteristic polynomial of the matrix has a zero eigenvalue it reduces to a third-order polynomial and an analytical expression for the largest eigenvalue is thus easily found [23]. The dimension of the set of reachable states is therefore

$$d_c(P(\Phi_\infty))=\frac{\log\left[\left[\frac{25}{54}+\sqrt{\frac{23}{108}}\right]^{1/3}+\left[\frac{25}{54}-\sqrt{\frac{23}{108}}\right]^{1/3}+\frac{2}{3}\right]}{\log(2)}\simeq 0.811\,37\ . \tag{3}$$

A consequence of this result is that the ratio of reachable states tends to zero as the lattice size of a cellular automaton under the action of rule 129 tends to infinity.

This method, of course, applies to all elementary cellular-automata rules and is straightforward once the local box definitions are given. The algorithm is as follows. Generate the system of linear difference equations governing the contents of the rows of Ψ_n. Compute the largest eigenvalue of the linear equation system and use this value to compute the capacity dimension. A complete table of the capacity dimension for 88 basic cellular automata rules is given in Table III.

The above method is equivalent to the method given by Wolfram [24]. In [24] the appropriate characteristic po-

TABLE III. Fractal dimensions of the set of reachable states of a basic set of cellular automata.

Rule	d_c	Rule	d_c	Rule	d_c	Rule	d_c
0	0	26	0.915 73	56	0.879 14	136	0.811 37
1	0.694 24	27	0.879 14	57	0.918 68	137	0.923 85
2	0.551 46	28	0.879 14	58	0.900 53	138	0.811 37
3	0.811 37	29	0.879 14	60	1	140	0.900 53
4	0.694 24	30	1	72	0.811 37	142	0.879 14
5	0.811 37	32	0.694 24	73	0.915 71	146	0.915 71
6	0.825 64	33	0.835 6	74	0.915 71	150	1
7	0.900 53	34	0.694 24	76	0.879 14	152	0.919 6
8	0.551 46	35	0.900 53	77	0.835 6	154	1
9	0.818 95	36	0.811 37	78	0.900 53	156	0.918 68
10	0.694 24	37	0.937 54	90	1	160	0.811 37
11	0.811 37	38	0.879 14	104	0.938 97	161	0.912 74
12	0.694 24	40	0.825 64	105	1	162	0.835 6
13	0.835 6	41	0.918 15	106	1	164	0.937 54
14	0.879 14	42	0.879 14	108	0.900 53	168	0.900 53
15	1	43	0.879 14	128	0.694 24	170	1
18	0.811 37	44	0.879 14	129	0.811 37	172	0.879 14
19	0.811 37	45	1	130	0.818 95	178	0.835 6
22	0.938 97	46	0.694 24	131	0.923 85	184	0.879 14
23	0.835 6	50	0.879 14	132	0.835 6	200	0.811 37
24	0.694 24	51	1	133	0.912 74	204	1
25	0.919 6	54	0.900 53	134	0.918 15	232	0.835 6

lynomials are derived from adjacency matrices of state transition graphs for minimal deterministic finite automata representing regular languages generated after one time step in the evolution of the cellular automata.

B. Dimension of the set of fixed points

A quantity influencing the dynamics of a cellular automaton is the number of fixed points. Since this number may vary with the size of the cellular automaton, an expression for the number of fixed points as a function of the lattice size is of interest. This subject has been thoroughly investigated by Jen [13], and it turns out that the number of fixed points F_n of a size n elementary cellular automaton can be expressed by a generalized Fibonacci recurrence relation $F_{n+1}=a_1F_n+a_2F_{n-1}$, where a_1 and a_2 are non-negative integers. We shall here present an alternative method based on the recursive box definitions where the same relations can be derived as a by-product of calculating the dimension of the set of fixed points.

The fixed points of any cellular automaton are represented by the boxes in Φ_n placed on the diagonal that goes through the origin of the global mapping. It is possible to calculate the fractal dimension of the fixed points using a technique similar to that applied in calculating the fractal dimension of reachable states in Sec. V A, since the fractal dimension of the diagonal of Ψ^n is identical to that of the diagonal of Φ^n, as n tends to infinity, provided that the selection rules do not cancel all the boxes of Ψ_n on the diagonal.

Using a rule-12 cellular automaton as an example, this is now demonstrated. The local box definitions of this rule is seen in Fig. 13. Since **B** and **C** are the only box types that appear after Ψ_1, we can describe the dynamics of the box types on the diagonal of Ψ_n in the matrix equation (4), which is readily derived from the box definitions in Fig. 13,

$$\begin{bmatrix}\mathbf{B}_{n+1}\\ \mathbf{C}_{n+1}\end{bmatrix}=\begin{bmatrix}0 & 1\\ 1 & 1\end{bmatrix}\begin{bmatrix}\mathbf{B}_n\\ \mathbf{C}_n\end{bmatrix}. \tag{4}$$

Here $\mathbf{B}_n$ and $\mathbf{C}_n$ are the number of **B** and **C** boxes, respectively, on the diagonal of Ψ_n. Since the largest eigenvalue of the matrix in (4) is $\lambda=(1+\sqrt{5})/2$, the dimension of the set of fixed points of Ψ_∞ is $d_c=\log(1+\sqrt{5})/\log(2)-1\simeq 0.694\,24$.

It follows readily from (4) that the relationship

$$F_{n+1}=F_n+F_{n-1} \tag{5}$$

governs the number of fixed points of Ψ_n. To obtain a similar expression for the fixed points of the exact mapping Φ_n, one has to consult the selection rules mentioned in Sec. IV A. The selection rules of a given column depends only on the first two and the last two bits of the corresponding bit sequence. Thus, for $n\geq 4$ the selection rules cancel a certain constant ratio of the boxes on the diagonal of Ψ_n. Unless the selection rules disqualifies all boxes on the diagonal of Ψ_n they will therefore have no effect on the relationship (5).

In the case of rule 12, it is easily seen that the selection rules will not rule out all the boxes on the diagonal of Ψ_n,

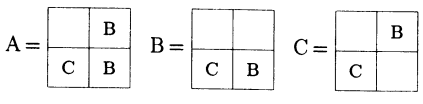


FIG. 13. Complete set of local box definitions for the rule-12 cellular automaton.

since all the columns in the first and the second main column of Ψ_n only contains one box and the majority of the boxes on the diagonal of Ψ_n are placed in this part of Ψ_n. This implies that equation (5) is valid for the number of fixed points of a rule-12 cellular automaton, as noted by Jen [13].

C. State orderings and properties of the mappings in the continuum limit

The basic method proposed in this paper is based on a graphical representation of the global mapping, in which the possible states of the cellular automaton are ordered according to the decimal equivalents of the binary sequences representing the lattice configuration. This ordering has been chosen due to its straightforwardness. Naturally, other orderings than the chosen one can be used, leading to different graphical representations of the global mappings, and hence different recursive definitions. However, the main principles of the method are not influenced by the ordering of states, as long as the ordering is based solely on the binary sequences representing the states of the cellular automaton. One may consider orderings taking into account the dynamics of the cellular automatons, and/or the rule table, but such orderings will be more complicated and will not necessarily be independent of the lattice size.

Due to periodic boundary conditions a change of the least significant bit of the state of a cellular automaton a may result in a change of the most significant bit in $b=\Phi(a)$. Also the two least significant bits of b may change. When the most significant bit of b depends on the least significant bit of a, a characteristic double structure of Φ is obtained (see, for instance, Fig. 2). This double structure can be avoided by using the updating scheme $a_i^{t+1}=\phi(a_i^t a_{i+1}^t a_{i+2}^t)$ instead of $a_i^{t+1}=\phi(a_{i-1}^t a_i^t a_{i+1}^t)$. In this way only the three least significant bits of $b=\Phi(a)$ depends on the least significant bit of a. The metric $d(a,b)=\sum_{i=1}^{n}|a_i-b_i|2^{-i}$ makes all cellular automata defined through the new updating scheme continuous for $n\to\infty$. The two updating schemes are of course equivalent, since their images can be transformed into each other via a simple shift of the periodic lattice. Due to the shifts of sub-boxes and the ordering of states, the graphical representation of the global mapping for n sufficiently large appears to be the picture of a discontinuous function. This applies to all local updating scheme. The apparent discontinuities are not only caused by long shifts. Short shifts and local subboxes has the same effect, even though they do not give rise to a double structure. In fact there is, in general, no well-defined limit function for $n\to\infty$. The discontinuities mentioned above presents the existence of such a unique limit function for almost all states and for almost all rules tables. In fact, out of the 256 elementary cellular automata rules, only rules 0, 51, 204, and 255 yields well-defined limiting functions. Note that, in principle, the global mappings for all rules can be transformed into C^1 functions of the interval [0;1] by using a recently proposed method by Moore [25].

VI. DISCUSSION

We have presented a method for recursive definition of the global mapping Φ_n for an elementary cellular automaton with n lattice sites, where $2\le n<\infty$. In the present paper we have only treated elementary cellular automata, but the method applies to any local cellular automaton.

By the use of nonlocal box definitions, which were introduced in Sec. II, the function $\mathbf{F}$: $\Phi_n\to\Phi_{n+1}$ can be defined from the rule table of any cellular automaton. Once the box definitions together with the global mapping of the two lattice site version of the cellular automaton are given, it is possible to update the cellular automaton. The computational resources needed for such an updating are proportional to the number of lattice sites and are therefore of the same order of magnitude as the computational resources required to update a cellular automaton in the traditional way, using the rule table. However, the real force of the recursive definition lies in the information it offers about the global state space structure of cellular automata. For instance, the self-similar state space structure of all cellular automata follows readily from the recursive definition of the global mapping.

A useful supplement to the global mapping Φ_n is the approximation Ψ_n, introduced in Sec. III. By use of Ψ_n it is possible to account for the apparent lattice size invariance of graphical representations of global mappings of cellular automata. Ψ_n is defined by the use of local box definitions, which can be calculated from the nonlocal box definitions defining Φ_n. The local box definitions can also be extracted directly from a graphical representation of the global mapping, by the use of the simple, geometrical method shown in Sec. IV. Due to the simplicity of local box definitions as opposed to nonlocal box definitions it is possible to calculate properties of Ψ_n, which are difficult to find for Φ_n. This was used to calculate the fractal dimension of the set of reachable states and the fractal dimension of the set of fixed points in cellular automata in the infinite lattice size limit (Secs. V A and V B).

Thus we have introduced a method based on recursive definitions of global mappings for cellular automata and applied it on a few problems concerning the global state space structure of cellular automata. The method is efficient for solving this kind of problems and offers a complementary geometrical approach to the solution methods proposed by Wolfram [24] and Jen [11].

There are several directions in which this global geometric approach can be taken. Most questions about the relation between the global state space structure and the dynamics of cellular automata are still open. An obvious next step is therefore to investigate higher order iterates of Φ_n and Ψ_n. At this point we have not made any systematic studies of these higher-order cellular-automata global mappings, but it is clear that the complexity of the cellular-automata dynamics is reflected by the higher iterates of Φ_n and Ψ_n. Recall that this is not the case for the first-order iterates.

Investigating higher-order iterates of Ψ_n and Φ_n is

closely related to the investigation of cellular automata with neighborhood radii larger than one, since a cellular automaton with neighborhood radius m can simulate m iterations of an elementary cellular automaton. Even though cellular automata with neighborhood radii larger than one complicates the box definitions somewhat, the method presented also directly applies to these automata.

Another question is whether symbolic dynamics can be applied successfully to global mappings of cellular automata. Following itineraries has been quite successful as a tool in the investigation of one-dimensional mappings [26]. Symbolic dynamics can for instance be applied to rule 170 with a fixed left boundary, and it can easily be shown that the dynamics of this cellular automaton is topologically conjugate to the shift map and is thus chaotic [27]. This method would facilitate a connection between the structure of the global mappings and the dynamical properties of cellular automata.

ACKNOWLEDGMENTS

We are grateful for the many enlightening discussions in particular with Mats Nordahl, James Theiler, and Chris Moore. Mats Nordahl and James Theiler are also acknowledged for their corrections and many constructive suggestions in earlier versions of this paper.

APPENDIX: DERIVATION OF LOCAL BOX DEFINITIONS FROM NONLOCAL BOX DEFINITIONS

In the following we describe the derivation of local box definitions from the nonlocal box definitions and the rule table. Since the nonlocal box definitions are generated from the rule table, the procedure illustrates how local box definitions can be derived directly from the rule table.

Assuming that the nonlocal box definitions are given in the way shown in Fig. 6, i.e., by the definitions of the boxes A through P, the local box definitions are derived in the following way: all examples used in the procedure correspond to a rule-137 cellular automaton.

For each nonlocal box definition $X \in \{A, B, \ldots, P\}$, one or two local boxes are defined.

In the case where one of the sub-boxes of X is shifted locally, two local box definitions $\mathbf{X}_1$ and $\mathbf{X}_2$ are made, one for each sub-box of X. (Note that local box definitions are written in boldface, while nonlocal definitions are written in italics.) Each of the boxes $\mathbf{X}_1$ and $\mathbf{X}_2$ is to contain one sub-box at the same position as the corresponding sub-box in X. The names of the sub-boxes in $\mathbf{X}_1$ and $\mathbf{X}_2$ should be the same as the names of the corresponding sub-boxes in X, with an index 1 added. For instance, the nonlocal box definition

$J =$ 

of a rule-137 cellular automaton (see Fig. 6) leads to the local definitions

$\mathbf{J}_1 =$ 

and $\mathbf{J}_2 =$ 

.

In the case that no sub-boxes are locally shifted, only one local box $\mathbf{X}_1$, containing the same sub-boxes as X, is defined. The names of the sub-boxes are to be the same as the corresponding sub-boxes of X, with index 1 added. For instance, the nonlocal box definition

$K =$ 

leads to the local definition

$\mathbf{K}_1 =$ 

.

Now all the local boxes defined so far are modified to take into account short shifts of sub-boxes. For each sub-box in the previously defined local boxes a new sub-box is added in case the nonlocal box definition corresponding to the sub-box contains short shifted sub-boxes. Each new sub-box gets the index 2 instead of 1 and it is placed right above or below (whichever is possible) the "parent" sub-box. As an example consider

$\mathbf{K}_1 =$ 

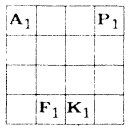
.

Since the nonlocal box definitions of J and I both contain short shifts, $\mathbf{K}_1$ is modified to

$\mathbf{K}_1 =$ 

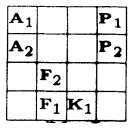
.

Next the positions of boxes in Ψ_2 are calculated. Place the local boxes $\mathbf{A}_1$, $\mathbf{F}_1$, $\mathbf{K}_1$, and $\mathbf{P}_1$ in Ψ_2 at the positions indicated by Φ_2 (the rule table is used to get this information), with $\mathbf{A}_1$ in the first column and so on. By consulting Φ_2 of a rule-137 cellular automaton we get



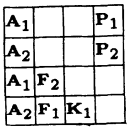
.

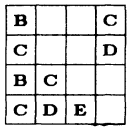
If the nonlocal box definition corresponding to a local box in Ψ_2 contains short shifted sub-boxes, a new box is added in Ψ_2, with the index 2 instead of 1. The new box is placed right above or below the original box in the same part (upper or lower) of Ψ_2. Since A, F, and P contain short shifts (see Fig. 6) we get



.

If the nonlocal box definition corresponding to a local box in Ψ_2 contains long shifted sub-boxes, the upper or lower part of the corresponding column in Ψ_2 is copied, so that the column contains the same structure in the upper part and in the lower part. Since A contains long shifts (see Fig. 6) we get

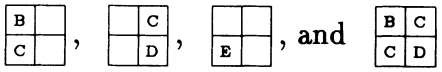




At this point there are 16 or more local box definitions. Since several of them are equivalent, the number of local box definitions is reduced by giving equivalent boxes the same name. By using the names **A,B,C,** . . . , i.e., no indices, a nice and simple notation is obtained. The names of the local boxes need not necessarily depend on the names of the corresponding nonlocal box definitions, since these are not referred to after this point in time. In case of rule 137, we end up with only four nonequivalent boxes corresponding the boxes **B, C, D,** and **E** in Fig. 10.

In order to obtain the definition of local boxes in Ψ_1 and Ψ_0, we divide Ψ_2 into four sub-boxes which are compared to the already defined local boxes. In the case that there are undefined sub-boxes, new local box definitions are added. The process is repeated for Ψ_0. In the case of a rule-137 automaton we have at this point

The four sub-boxes of Ψ_2 are



which are easily identified as the previously defined boxes **B, C, D,** and **E**. Finally we see that

$\Psi_0 =$ 

does not correspond to a previously defined box. Thus we assign the new **A** to it. With

A = 

we have a complete set of local box definitions corresponding exactly to Fig. 10.

[1] U. Frish, B. Hasslacher, and Y. Pomeau, Phys. Rev. Lett. **56**, 1505 (1986).

[2] *Lattice Gas Methods for Partial Differential Equations,* edited by G. Doolen, U. Frisch, B. Hasslacher, S. Orszag, and S. Wolfram, Santa Fe Institute, Studies in the Sciences of Complexity Vol. IV (Addison-Wesley, Redwood City, CA, 1990).

[3] Lattice Gas Methods for PDE's: Theory, Applications, and Hardware, edited by G. Doolen [Physica D **47**, (1991)].

[4] *Numerical Methods for the Simulation of Multi-Phase and Complex Flow,* edited by T. M. Verheggen, Lecture Notes in Physics Vol. 398 (Springer-Verlag, Berlin, 1992).

[5] Advanced Research Workshop on Lattice Gas Automata Theory, Implementation and Simulation, edited by J. P. Boon [J. Stat. Phys. **68**, (3/4) (1992)].

[6] *Microscopic Simulations of Complex Hydrodynamics Phenomena, Vol. 292 of NATO Advanced Study Institute, Series B: Physics,* edited by M. Mareschal and B. L. Holian (Plenum, New York, 1992).

[7] *Artificial Life,* edited by C. Langton, Santa Fe Institute, Studies in the Sciences of Complexity Vol. IV (Addison-Wesley, Redwood City, CA, 1989).

[8] *Artificial Life II,* edited by C. Langton, C. Tayler, D. Farmer, and S. Rasmussen, Santa Fe Institute, Studies in the Sciences of Complexity Vol. X (Addison-Wesley, Redwood City, CA, 1991).

[9] J. von Neumann, in *Theory of Self-Reproducing Automata,* edited by A. Burks (University of Illinois Press, Urbana, IL, 1966).

[10] *Theory and Applications of Cellular Automata,* edited by S. Wolfram (World Scientific, Singapore, 1986).

[11] E. Jen, in *Lectures in the Sciences of Complexity,* edited by D. L. Stein (Addison-Wesley, Reading, MA, 1989), pp. 743–758.

[12] E. Jen, Nonlinearity **4**, 251 (1990).

[13] E. Jen, Commun. Math. Phys. **118**, 569 (1988).

[14] K. Lindgren and M. G. Nordahl, Complex Syst. **2**, 409 (1988).

[15] M. G. Nordahl, Complex Syst. **3**, 63 (1989).

[16] *Cellular Automata: Theory and Experiment,* edited by H. Gutowitz [Physica D **45** (1990)].

[17] S. Wolfram, Rev. Mod. Phys. **55**, 601 (1983).

[18] W. Li and N. Packard, Complex Syst. **4**, 281 (1990); **5**,

[19] C. Knudsen and R. Feldberg, M. Sc. thesis, The Technical University of Denmark (in Danish), 1991.

[20] G. A. Edgar, *Measure, Topology and Fractal Geometry* (Springer-Verlag, Berlin, 1990).

[21] C. H. Bennett, in *Emerging Syntheses in Science,* edited by D. Pines (Addison-Wesley, Redwood City, CA, 1989).

[22] C. H. Bennett, in *Lectures in the Sciences of Complexity,* edited by D. L. Stein (Addison-Wesley, Redwood City, CA, 1989), Vol. I.

[23] *Handbook of Mathematical Functions,* edited by M. Abramowitz and I. A. Stegun (Dover, New York, 1964).

[24] S. Wolfram, Commun. Math. Phys. **96**, 15 (1984).

[25] C. Moore (unpublished).

[26] P. Collet and J.-P. Eckmann, *Iterated Maps of the Interval as Dynamical Systems* (Birkhäuser, Boston, 1980).

[27] R. L. Devaney, *Chaotic Dynamical Systems* (Addison-Wesley, Redwood City, CA, 1989).